

### **DETAILED ACTION**

This office correspondence is response to the applicant's amendment filed on March 12, 2009.

Claims 1 – 5, 7 – 13, and 15 - 31 are pending.

Claims 6 and 14 are cancelled.

### **EXAMINER'S AMENDMENT**

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Jonathan S. Miller, Registration No. 48,534 on 04/22/2008.

The applicant has been amended as follows:

Claim 1, A method comprising: establishing communication between a plurality of non-Java-based server node and a plurality of Java-based server nodes via an intermediate server, wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

generating a packet, on a first non-Java-based server nodes, to be transmitted from one-of-the first non-Java-based server nodes to a first Java-based server nodes;

specifying in a header of the packet an address of the first Java-based server node and information that indicates that the packet is generated by the first non-Java-based server nodes;

forwarding the packet to the intermediate server from the first non-Java-based server nodes;

forwarding the packet to first Java-based server node from the intermediate server based on the address provided in the header of the packet, wherein the packet header is formatted to be compatible with the format of the first Java-based server node such that it can be decoded by the first Java-based server node, wherein the intermediate server providing interoperability of communications between the Java based server nodes and the non-Java based server nodes.

maintaining a list of services, the list of services includes processes and tasks performed by the Java-based server nodes;

and maintaining a list of services, the list of services includes processes and tasks performed by the non-Java-based server nodes, wherein the maintained lists of services facilitate communications between the Java-based server nodes and the non-Java-based server nodes.

Claim 2, The method of claim 1, further comprising: generating a second packet to be transmitted from a second Java-based server node to a second non-Java-based server node;

specifying in a header of the second packet an address of the second non- Java-based server node and information that indicates that the packet is generated by the second Java-based server node;

forwarding the second packet to the intermediate server from the second Java-based server node;

and forwarding the second packet to the second non-Java-based server node from the intermediate server based on the address provided in the header of the second packet.

Claim 3, The method of claim 1, further comprising: sending notification of a status of each of the listed services to the non-Java-based server nodes such that the maintained lists of services can be updated

Claim 4, The method of claim 1, further comprising: sending notification of a status of each of the listed services to the Java-based server nodes such that the maintained lists of services can be updated.

Claim 5, The method of claim 1, wherein the maintaining a list of services is accomplished by the intermediate server and the sending notification of a status of each of the listed services is accomplished by the intermediate server.

Claim 6, Cancelled

Claim 7, A system comprising: a plurality of non-Java-based server nodes, each of the non- Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from first non- Java-based server nodes wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

a plurality of Java-based server nodes, each of the Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from a first Java-based server node;

a message server coupled between the Java-based server nodes and the non-Java-based server nodes to establish communication between the first Java-based server node and the first non-Java-based server node, by formatting each packet header such that it is compatible with a format of a destination server.;

and an enqueue server coupled between the Java-based server nodes and the non-Java based server nodes to provide central locking services to lock access to resources in the system for use during communications between the first Java-based server node and the first non-Java based server node such that communications between the first Java-based server node and the first non- Java based server node are not interrupted.

Claim 8, The system of claim 7, further comprising a dispatcher to distribute client requests for services to the Java-based server nodes and the non-Java based server nodes, such that the dispatcher distributes the load amongst the Java-based server nodes and the non-Java based server nodes during communications.

Claim 9, The system of claim 7, wherein the message server provides communications between the Java-based server nodes and the non-Java based server nodes by routing message packets between the non-Java-based server nodes and the Java-based server nodes

Claim 10, The system of claim 7, wherein the message server is to assign a service identification associated with each type of services executed on the server nodes, wherein assigning the service identifications assists in providing communications between the Java-based server nodes and the non-Java based server nodes.

Claim 11, The system of claim 10, wherein the message server includes a service repository to maintain a list of the assigned service identification and corresponding service names, wherein maintaining the service identifications and the service names assists in providing communications between the Java-based server nodes and the non-Java based server nodes.

Claim 12, The system of claim 8, wherein the message server further comprises:  
a first repository to maintain a list of services currently being executed on the non-Java-based server nodes;

and a second repository to maintain a list of services currently being executed on the Java- based server nodes, wherein the first and second repositories facilitate distribution of load by indicating currently executing services.

Claim 13, The system of claim 8, wherein the message server is to maintain a list of services performed by the Java-based server nodes and the non-Java based server nodes and a status corresponding to each of the listed services, and to send notification of the status of the listed services dispatcher such that the dispatcher distributes the load in each instance following requests for services.

Claim 14, Cancelled

Claim 15, A message server comprising: a first communication interface to establish communication with a plurality of non-Java- based server nodes;

a second communication interface to establish communication with a plurality of Java- based server nodes, wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

and a controller to transfer packets between the non-Java-based server nodes and the Java- based server nodes, the controller to ensure the packets are received by

a destination server node by resending the packets if a confirmation of receipt has not been received from the destination server node.

Claim 16, The message server of claim 15, wherein the controller is to assign a service identification associated with each type of services executed on the Java based server nodes and non-Java-based server nodes, wherein assigning the service identifications assists in providing communications between the first and second communications interfaces.

Claim 17, The message server of claim 16, further comprising: a service repository maintain a list of the assigned service identification and corresponding service names, wherein maintaining the service identifications and the service names assists in providing communications between the first and second communications interfaces.

Claim 18, The message server of claim 15, further comprising: a first repository to maintain a list of services currently being executed on the non-Java- based server nodes;

and a second repository to maintain a list of services currently being executed on the Java- based server nodes, wherein the first and second repositories facilitate distribution of load by indicating currently executing services.

Claim 19, The message server of claim 17, wherein the controller is to send notification of a status of each of the services listed in the first repository to the non-Java-based server nodes such that the maintained lists of services can be updated.

Claim 20, The message server of claim 17, wherein the controller is to send notification of a status of each of the services listed in the second repository to the Java-based server nodes such that the maintained lists of services can be updated.

Claim 21, A machine-readable medium that provides instructions, which when executed by a processor cause the processor to perform operations comprising:

establishing communication with a plurality of non-Java-based server nodes;

establishing communication with a plurality of Java-based server nodes, wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

transferring packets between the non-Java-based server nodes and the Java-based server nodes;

and ensuring the packets are received by a destination server node by resending the packets if a confirmation of receipt has not been received from a destination server node.

Claim 22, The machine-readable medium of claim 21, wherein the operations performed by the processor further comprise: assigning a service identification associated with each type of services executed on the server nodes;

and maintaining a list of the assigned service identification and corresponding service names, wherein maintaining the service identifications and the service names assists in providing communications between the non-Java-based server nodes and the Java-based server nodes.

Claim 23, The machine-readable medium of claim 21, wherein the operations performed by the processor further comprise: maintaining a list of services currently being executed on the non-Java-based server nodes in a first repository;

and maintaining a list of services currently being executed on the Java-based server nodes in a second repository, wherein each list facilitates distribution of load by indicating currently executing services.

Claim 24, The machine-readable medium of claim 23, wherein the operations performed by the processor further comprise: sending notification of a status of each of the services listed in the first repository to the non-Java-based server nodes such that the maintained lists of services can be updated;

and sending notification of a status of each of the services listed in the second repository to the Java-based server nodes such that the maintained lists of services can

be updated.

Claim 25, A system comprising: means for generating a packet on a first non-Java based server node such that a header of the packet specifies an address of a first Java-based server node wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

means for indicating that the packet is generated by a non-Java based server node;

means for forwarding the packet to intermediate communication means from the first non-Java-based server node, wherein the intermediate communication means provides interoperability of communications between the Java based server nodes and the non-Java based server nodes;

and means for forwarding the packet to the first Java-based server node from the intermediate communication means based on the destination address provided in the header of the packet;

means for maintaining a list of services, the list of services includes processes and tasks performed by the first Java-based server node, wherein maintaining the list of service identifications and the service names assists in providing communications between the first non- Java-based server node and the first Java-based server node;

and means for sending notification of a status of each of the listed services to the first Java- based server node, the notification indicates whether the service is running or

stopped such that the maintained list of services can be updated.

Claim 26, The system of claim 25, further comprising: means for generating a second packet on a second Java based server node such that a header of the second packet specifies an address of a second non-Java-based server node and that the second packet originated from a Java-based server nodes;

means for forwarding the second packet to the intermediate communication means from second Java-based server nodes;

and means for forwarding the second packet to the second non-Java-based server node from the intermediate communication means based on the destination address provided in the header of the second packet.

Claim 27, The system of claim 26, wherein the intermediate communication means further comprises: means for maintaining a list of services performed by the Java-based server nodes wherein maintaining the list assists in providing communications between the Java-based server nodes and the non-Java-based server nodes; and means for sending notification of a status of each of the listed services to the Java-based server nodes such that the maintained list of services can be updated.

Claim 28, The system of claim 27, wherein the intermediate communication means further comprises: means for maintaining a list of services performed by the non-Java-based server nodes wherein maintaining the list of service identifications and the

service names assists in providing communications between the non-Java-based server nodes and the Java-based server nodes;

and means for sending notification of a status of each of the listed services to the non-Java- based server nodes such that the maintained list of services can be updated.

Claim 29, The system of claim 25, wherein the intermediate communication means further comprises: means for establishing communication with a plurality of non-Java-based server nodes;

means for establishing communication with a plurality of Java-based server nodes;

and means for transferring packets between the non-Java-based server nodes and the Java- based server nodes.

Claim 30, The system of claim 25, wherein the intermediate communication means further comprises: means for assigning a service identification associated with each type of services executed on the Java based server nodes and the non-Java based server nodes;

and means for maintaining a list of the assigned service identification and corresponding service names, wherein maintaining the service identifications and the service names assists in providing communications between the Java-based server nodes and the non-Java-based server nodes.

Claim 31, A system comprising: a plurality of non-Java-based server nodes, each of the non-Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from a first non-Java-based server node;

a plurality of Java-based server nodes, each of the Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from a first Java-based server node, wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications;

a message server coupled between the Java-based server nodes and the non-Java-based server nodes to establish communication between the first Java-based server node and the first non-Java-based server node by formatting each packet header such that it is compatible with a format of a destination server, the message server comprising:

a first repository to maintain a list of processes and tasks performed by the plurality of non-Java based server nodes, the first repository includes a plurality of rows each associated with a service performed by a non-Java based server and a plurality of columns which identify attributes associated with each service, the attributes include a server identification to indicate a server performing the service, a service mask to identify the type of service, and a status to indicate the status of the service;

a second repository to maintain a list of processes and tasks performed by the plurality of Java based server nodes, the second repository includes a plurality of rows

each associated with a service performed by a Java based server and a plurality of columns which identify attributes associated with each service, the attributes include a server identification to indicate a server performing the service, a service mask to identify the type of service, and a status to indicate the status of the service;

and a third repository to maintain a list of assigned service identifications and their corresponding service names, the third repository includes a plurality of rows each associated with a service and a plurality of columns which identify attributes associated with each service, the attributes include a service name to indicate the name of the service and an assigned service mask to indicate the type of service.

### **REASONS FOR ALLOWANCE**

The following is the examiner's statement of reasons for allowance:

There is no teaching or suggestion in the prior art of the combination of claim limitations, among other things, as presented in claims 1 - 31, specifically that a communication between a plurality of non-Java-based server node and a plurality of Java-based server nodes via an intermediate server wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications.

This is found in Applicant's specification on page 5 Paragraph 5.

The closest prior art found by the Examiner is the previously cited Martin et al., and Harris et al; however, this combination of references does not teach or suggest a communication between a plurality of non-Java-based server node and a plurality of Java-based server nodes via an intermediate server wherein Java-based server nodes are server nodes that implement java 2 Platform Enterprise Edition (J2EE) applications. For these reasons, in conjunction with the other claim limitations puts this case in condition for allowance.

### CORRESPONDANCE INFORMATION

Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Shaq Taha** whose telephone number is 571-270-1921. The examiner can normally be reached on 8:30am-5pm Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, **Jeff Pwu** can be reached on 571-272-6798.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/S. T./  
Examiner, Art Unit 2446

/Jeffrey Pwu/  
Supervisory Patent Examiner, Art Unit 2446

